Latent Clustering on Graphs with Multiple Edge Types [⋆]

Matthew Rocklin[1] and Ali Pinar[2]

[1] mrocklin@cs.uchicago.edu Department of Computer Science, University of Chicago
[2] apinar@sandia.gov Sandia National Laboratories

Abstract. We study clustering on graphs with multiple edge types. Our main motivation is that similarities between objects can be measured in many different metrics, and so allowing graphs with multivariate edges significantly increases modeling power. In this context the clustering problem becomes more challenging. Each edge/metric provides only partial information about the data; recovering full information requires aggregation of all the similarity metrics. We generalize the concept of clustering in single-edge graphs to multi-edged graphs and discuss how this generates a space of clusterings. We describe a meta-clustering structure on this space and propose methods to compactly represent the meta-clustering structure. Experimental results on real and synthetic data are presented.

1 Introduction

Graphs are widely recognized as the standard modeling language to represent relations between entities of a complex system. Entities in the data are represented as nodes while relationships between entities are represented as edges between nodes. For instance, an email network would have email accounts as nodes, and the email exchanges between two accounts form an edge between the two nodes. Proteins (nodes) are connected in a protein interaction network by an edge if the proteins are part of the same system function.

In many real-world problems, connections or similarities between entities can be defined by many different relationships, where connections/similarities are quantified by boolean (a connection exists or not), or continuous variables. For example, similarity between two scientific articles can be defined based on authors, citations to, citations from, keywords, titles, where they are published, text similarity, etc.... Relationships between people can be based on the nature of the relationship (e.g., business, family, friendships) or the means of communication (e.g., email, phone, personal meetings). Electronic files can be grouped by their type (Latex, C, html), names, the time they are created, or the pattern they are accessed. In these examples, there are multiple graphs that define relationships between the subjects. In sociology these graphs are called "graphs with

[⋆] This work was funded by the applied mathematics program at the United States Department of Energy and performed at Sandia National Laboratories, a multiprogram laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

multiple relations, multivariate graphs, or multiplexed graphs."[5] For brevity we use "multiweighted graphs." These multiweighted graphs differ from traditional multigraphs. In our case we have a fixed number of labeled edges rather than a multigraph which has a variable number of unlabeled edges.

This paper studies the community detection problem on networks with multiple edges-types/relations. Clustering is a method to reduce the complexity of a singly-weighted graph while still retaining much of its information. Groups of vertices (clusters) are formed which are well connected within the cluster and sparsely connected between clusters. This technique is a critical enabler in unsupervised machine learning and continues to be a very active area of research. Almost all methods however, require a singly-weighted graph. It is convenient to aggregate multi-weighted edges to a single composite edge. However, the choice of the aggregation function should be done cleverly, and we should be able to analyze the inevitable loss of information in the results.

Consider the situation where several edge types share redundant information yet as an ensemble combine to form some broader structure. For example scientific journal articles can be connected by text similarity, abstract similarity, keywords, shared authors, cross-citations, etc.... Many of these edge types reflect the *topic* of the document while others are also influenced by the *location* of the work. Text, abstract, and keyword similarity are likely to be redundant in conveying topic information (physics, math, biology) while shared authorship (two articles sharing a common author) is likely to convey both topic and location information because we tend to work with both those in our same field and with those in nearby institutions. We say that the topic and location attributes are *latent* because they do not exist explicitly in the data. We can represent much of the variation in the data by two relatively independent clusterings based on the topic of documents and their location. This compression of information from five edge types to two meaningful clusterings is the goal of this paper.

1.1 Contributions

The community detection problem on networks with multiple edge types bears many interesting problems. In our earlier work we studied how to compute an aggregation scheme that best resonates with the ground-truth data, when such data was available [12]. In this work we study the following questions: Is there a meta-clustering structure, (i.e., are the clusterings clustered) and if so how do we find it? How do we find significantly different clusterings for the same data? Our main contributions in this paper are as follows.

- We describe how the space of clusterings can be searched using sampling methods, and investigate the structure of this space. We introduce the meta-clusters: while the clusterings vary with how we aggregate various similarity measures, these clusterings gather around a small number of clusters. That is clusterings are nicely clustered.

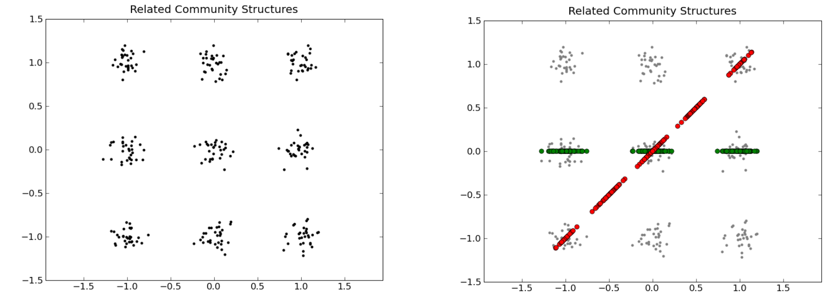
- We propose methods to efficiently represent the space of clusterings with minimal loss of information. More specifically, if we can produce a handful of clusterings that represent the meta-clusters, then these small number of clusters can be used for data analysis, providing a more accurate and thorough information of the data, at a reasonable increase in processing times.
- We apply our proposed techniques to a data set collected from scientific articles in the arXiv database, and show that or proposed techniques can be successfully adopted for analysis of real data.

1.2 An Illustrative Problem

We construct a simple multiweighted network to demonstrate latent classes. For illustration, we assume our graph is perfectly embedded in $\mathbb{R}^2$ as seen in Fig. 1a. In this example each point on the plane represents a vertex, and two vertices are connected by an edge if they are close in distance. The similarity/weight for each edge is inversely proportional to the Euclidean distance. We see visually that there are nine natural clusters. More interestingly we see that these clusters are arranged symmetrically along two axes. These clusters have more structure than the set $\{1, 2, 3, ..., 9\}$. Instead they have the structure $\{1, 2, 3\} \times \{1, 2, 3\}$. An example of such a structure would be the separation of academic papers along two factors, {Physics, Mathematics, Biology} and {West Coast, Midwest, East Coast}. The nine clusters (with examples like physics articles from the West or biology articles from the Midwest) have underlying structure.

Our data sets do not directly provide this information. For instance with journal articles we can collect information about authors, where the articles are published, and their citations. Each of these aspects provides only a partial view of the underlying structure. Analogous to our geometric example above we could consider features of the data as projections of the points to one dimensional subspaces. Distances/similarities between the points in a projection have only partial information. This is depicted pictorially in Fig. 1b. For instance, the green projection represents a metric that clearly distinguishes between columns but cannot differentiate between different communities on the same column. The red projection on the other hand provides a diagonal sweep, capturing partial information about columns and partial information about rows. Neither of the two metrics can provide the full information for the underlying data. However when considered as an ensemble they do provide a complete picture. Our goal is to be able to tease out the latent factors of data from a given set of partial views.

In this paper, we will use this 3×3 example for conceptual purposes and for illustrations. Our approach is construct many multi-weighted graphs by using



(a) 270 vertices arranged in nine clusters on the plane. Edges exist between vertices so that close points are well connected and distant points are poorly connected.

(b) Two 1D graphs arranged to suggest their relationship to the underlying 3x3 community structure. Both have clear community structures that are related but not entirely descriptive of the underlying 3x3 communities.

Fig. 1. Illustrating clusters (a) underlying structure and (b) low-dimensiona/partial views

combinations of the partial views of the data. We will cluster these graphs and analyze these clusters to recover the latent structure.

2 Background

A weighted graph is represented as a tuple $G = (V, E)$, V a set of vertices and E a set of edges. Each edge e_i is a tuple $e_i = \{v_a, v_b, w_i \mid v_a, v_b \in V, w_i \in \mathbb{R}\}$ representing a connection between vertices v_a and v_b with weight w_i. In this work we replace $w_i \in \mathbb{R}$ with $\boldsymbol{w_i} \in \mathbb{R}^k$ with k being the number of edge types. We will construct functions that map multiweighted edges $\boldsymbol{w_i} \in \mathbb{R}^k$ to *composite edge types* $f(\boldsymbol{w_i}) = \omega_i \in \mathbb{R}$. In this paper f will be linear $\omega_i = \sum \alpha_i w_i$.

2.1 Clustering

Intuitively, the goal of clustering is to break down the graph into smaller groups such that vertices in each group are tightly coupled among themselves and loosely coupled with the remainder of the network. Both the translation of this intuition into a well-defined mathematical formula and design of associated algorithms pose big challenges. Despite the high quality and the high volume of the literature, the area continues to draw a lot of interest due to the growing importance of the problem and the challenges posed by the size and mathematical variety of the subject graphs.

Our goal here is to extend the concept of clustering to graphs with multiple edge types without getting into the details of clustering algorithms and formulations, since such a detailed study will be well beyond the scope of this paper.

In this paper, we used *Graclus*, developed by Dhillon et al[3], which uses the top-down approach that recursively splits the graph into smaller pieces and *Fast-Community* developed by Clauset et al[2] which uses an agglomerative approach which optimizes the modularity metric. For further information on clustering see Lancichinetti et al.[6].

2.2 Variation of Information of Clusterings

At the core of most of our discussions will be similarity between two clusterings. Several metrics and methods have been proposed for comparing clusterings, such as *variation of information* [9], *scaled coverage measure* [13], *classification error* [7–9], and *Mirkin's metric* [10]. Out of these, we have used the variation of information metric in our experiments.

Let $C_0 = \langle C_0^1, C_0^2, \ldots, C_0^K \rangle$ and $C_1 = \langle C_1^1, C_1^2, \ldots, C_1^K \rangle$ be two clusterings of the same node set. Let n be the total number of nodes, and $P(C,k) = \frac{|C^k|}{n}$ be the probability that a node is in cluster C^k in a clustering C. Similarly the probability that a node is in cluster C^k in clustering C_i and in cluster C^l in clustering C_j is $P(C_i, C_j, k, l) = \frac{|C_i^k \cap C_j^l|}{n}$. The *entropy of information* or expectation value of learned information in C_i is defined

$$H(C_i) = -\sum_{k=1}^{K} P(C_i, k) \log P(C_i, k)$$

the mutual information shared by C_i and C_j is

$$I(C_i, C_j) = \sum_{k=1}^{K} \sum_{l=1}^{K\prime} P(C_i, C_j, k, l) \log P(C_i, C_j, k, l),$$

Given these two quantities Meila defines the variation of information metric by

$$d_{VI}(C_i, C_j) = H(C_i) + H(C_j) - 2I(C_i, C_j). \tag{1}$$

Meila [9] explains the intuition behind this metric a follows. $H(C_i)$ denotes the average uncertainty of the position of a node in clustering C_i. If, however, we are given C_j, $I(C_i, C_j)$ denotes average reduction in uncertainty of where a node is located in C_i. If we rewrite Equation (1) as

$$d_{VI}(C_i, C_j) = (H(C_i) - I(C_i, C_j)) \;\; + \;\; (H(C_j) - I(C_i, C_j)),$$

the first term measures the information lost if C_j is the true clustering and we know instead C_i, and the second term is the opposite.

The variation of information metric can be computed in $O(n)$ time.

2.3 Previous Work

Similar problems have been approached in previous work. Mucha et al.[11] looked at community detection when multiple edge types are sampled in time and strongly correlated. Dunlavy et al. [4] described this problem as a three dimensional Tensor and used a PARAFAC decomposition (SVD generalization) to identify dominant factors.

3 Searching the Space of Clusterings

From a multiweighted graph $G = (V, E)$ with edges $e_i \in E = (v_a, v_b, \langle w_i^0, w_i^1, \ldots, w_i^k \rangle)$ we can build a composite edge-type $\omega_i = \sum_j \alpha_j w_i^j$. This composite edge-type along with the vertex set V define a graph $G_{\boldsymbol{\alpha_j}}$ indexed by the vector α_j. We may apply a traditional clustering algorithm $\mathcal{C}$ to this graph to obtain a clustering $\mathcal{C}(G_{\boldsymbol{\alpha_j}}) = C_{\boldsymbol{\alpha_j}}$. This process identifies with each point $\boldsymbol{\alpha_j} \in \mathbb{R}^k$ a clustering $C_{\boldsymbol{\alpha_j}}$. Thus a multiweighted graph is imbued with a *space* of clusterings.

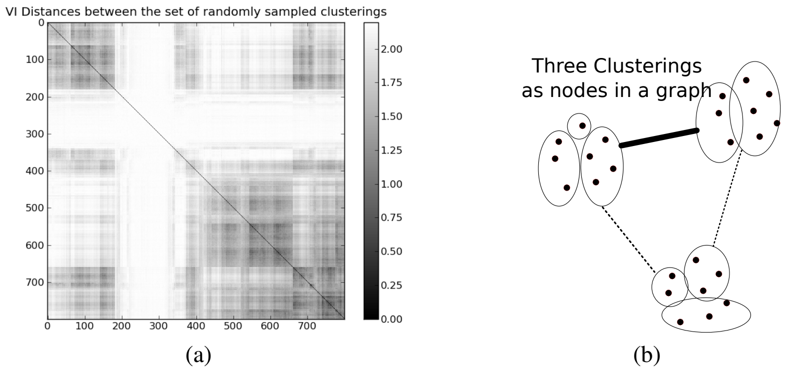
We expect that different regions of this space will have different clusterings. How drastic these differences are will depend on the particular multiweighted graph. How can we characterize this space of clusterings? Are there homogeneous regions, easily identifiable boundaries, groups of similar clusterings, etc...? We investigate the existence of a meta-clustering structure. That is we search for whether or not several clusterings in this space exhibit community structure themselves. In this section, we present our methods for these questions on the 3×3 data. We will later provide results on a larger data set.

3.1 Sampling the Clustering Space

To inspect the space of clusterings we sample in a Monte Carlo fashion. We take points $\boldsymbol{\alpha_i} \in \mathbb{R}^k$ such that $|\boldsymbol{\alpha_i}| = 1$, and compute the appropriate graph and clustering at each point. We may then compare these clusterings in aggregate.

As our first experiment, we take 16 random one-dimensional projections of the points laid out in the plane shown in Fig. 1 and consider the projected-point-wise distances in aggregate as a multiweighted graph. From this multiweighted graph we take 800 samples of the linear space of clusterings. These 800 clusterings approximate the clustering structure of the multiweighted graph.

The results of these experiments are presented in Figure 2(a). In this figure each row and column corresponds to a clustering of the graph. Entries in the matrix represent the variation of information distance between two clusterings. Therefore dark regions in this matrix are sets of clusterings that are highly similar. White bands show informational independence between regions. The rows/columns of this matrix have been ordered to have more similar clusterings closer to each other so as to highlight the clusters of clusterings detected.



(a)

(b)

Fig. 2. The Meta-clustering information (a) VI distances between 800 sampled clusterings. Vertices are ordered to show optimal clustering of this graph. Dark blocks on the diagonal represent clusters. The white band is a group of completely independent clusterings. (b) Three Clusterings treated as nodes in a graph. Similar clusterings (top two) are connected with high-weighted edges. Distant clusterings are connected with low-weighted edges.

3.2 Meta-clusters: Clusters of Clusterings

While it is interesting to know that significantly different clusterings can be found, the lack of stable clustering structure is not helpful for applications of clustering such as for unsupervised learning. We need to reduce this set of clusterings further. We approach this problem by applying the idea of clustering onto this set of clusterings. We call this problem the *meta-clustering* problem.

We represent the clusterings as nodes in a graph and connect them with edge-weights determined by the inverse of the variation of information metric [9]. We inspect this graph to see if it contains clusters. That is, we *cluster the graph of clusterings* to see if there exist some tightly coupled clusters of clusterings within the larger space. For instance in Fig. 2(b) the top two clusterings differ only in the position of a single vertex and thus are highly similar. In contrast the bottom clustering is different from both and is weakly connected.

Figure 2(a) reveals the meta-clustering structure in our experiments. The dark blocks around the diagonal correspond to meta-clusters. We can see two big blocks in the upper left and lower right corners. Furthermore, there is a hierarchical clustering structure within these blocks, as we see smaller blocks within the larger blocks. In this experiment, we were able to observe meta-clusters. As usual, results depend on the particular problem instance. While we do not claim that one can always find such meta-clusters, we expect that they will exist in many multi-weighted graphs, and exploiting the meta-clustering structure can enable efficiently handling this space, which is the topic of the next section.

4 Efficient Representation of the Clusterings

In this section we study how to efficiently represent the meta-clustering structure. First we will study how to reduce a cluster of clusterings into a single averaged or representative clustering. Then, we will study how to select and order a small number of meta-clusters to cover the clustering space efficiently.

4.1 Averaging Clusterings within a Cluster

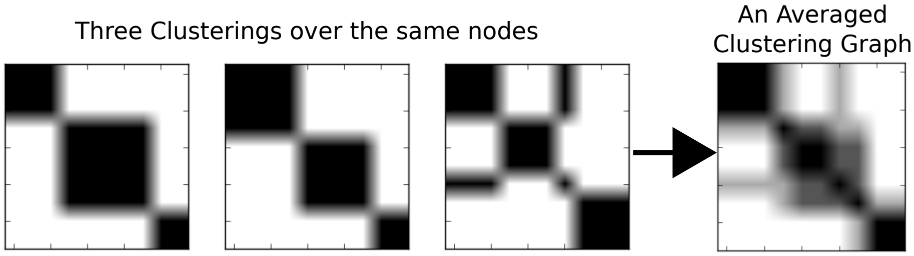


Fig. 3. Showing the CSPA [14] averaging procedure for clusterings. Each clustering is displayed as a block diagonal graph (or permutation) with two nodes connected if and only if they are in the same cluster. Then an aggregate graph (right) is formed by the addition of these graphs. This graph on the right is then clustered using a traditional algorithm. This clustering is returned as the representative-clustering.

To increase the human accessibility of this information we reduce each cluster of clusterings into a single representative clustering. We use the ”Cluster-based Similarity Partitioning Algorithm” (CSPA) proposed by Strehl et. al [14] to combine several clusterings into a single average. In this algorithm each pair of vertices is connected with an edge with weight equal to the number of clusters in which they co-occur. If v_a and v_b are in the same cluster in k of the clusterings then in this new graph they are connected with weight k. If they are never in the same cluster then they are not connected. We then cluster this graph and use the resultant clustering as the representative. In Fig. 3 we depict the addition of three clusterings to form an average graph which can then be clustered.

We perform this process on the clusters of clusterings found in section 3.2 and presented in Fig. 2(a) to obtain the *representative-clusterings* in Fig. 4. We see that the product of the first two representative-clusterings identifies the original nine clusterings with little error. We see also that the two factors are identified perfectly by each of these clusterings individually.

4.2 Ordering by Set-Wise Information Content

In Fig. 4, the original 3x3 community structure can be reconstructed using only the first two representative-clusterings. Why are these two chosen first? Select-

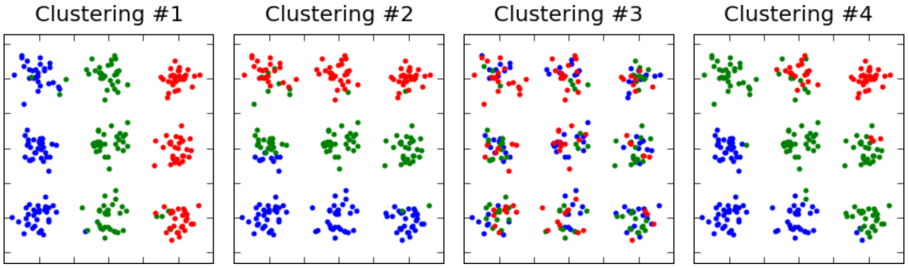


Fig. 4. Representative-Clusterings of the four dominant clusters-of-clusterings from Fig. 2(a). Clusterings are displayed as colorings of the original points in the 2-d plane. These are ordered to maximize cumulative set-wise information. Notice how the first two representative-clusterings recover the original nine clusterings exactly.

ing the third and fourth representative-clusterings would not have had this pleasant result. How should we order the set of representative-clusterings?

We may judge a set of representative-clusterings by a number of factors: *(i)* How many of our samples ascribe to the associated meta-clusters, what fraction of the space of clusterings do they cover? *(ii)* How much information do the clusterings cover as a set? *(iii)* How redundant are the clusterings? How much informational overlap is present? We would like to maximize information while minimizing redundancy. In Fig. 4 we ordered the representative-clusterings to maximize setwise information. Minimizing redundancy came as a fortunate side-effect. Notice how each of the clusterings in order is independent from the preceding ones. Knowing that a vertex is red in the first image tells you nothing about the color of the vertex in the second. The second therefore brings only novel information and no redundancy.

To compute the information content of a set of clusterings we extend the Variation of Information metric in a natural way. In section 2.2 we introduced the mutual information of two clusterings as follows:

$$I(C_i, C_j) = \sum_{k=1}^{K} \sum_{l=1}^{K\prime} P(C_i, C_j, k, l) \log P(C_i, C_j, k, l),$$

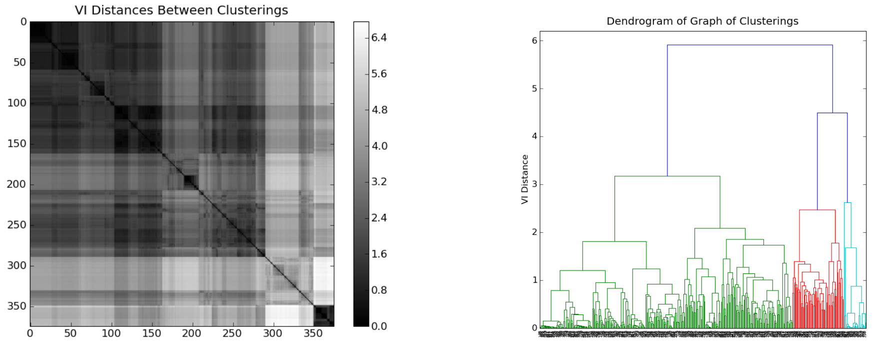
where $P()$ is the probability that a randomly selected node was in the specified clusters. This is equivalent to the self-information of the Cartesian product of the two clusterings. Its extension to a set of clusterings $I(C_\alpha, C_\beta, \ldots, C_\omega)$ is

$$\sum_{a=1}^{K} \sum_{b=1}^{K'} \ldots \sum_{z=1}^{K'''} P(C_\alpha, C_\beta, \ldots, C_\omega, a, b, \ldots, z) \log P(C_\alpha, C_\beta, \ldots, C_\omega, a, b, \ldots, z).$$

For a large number of clusterings or large K this quickly becomes inconvenient. In these cases we order the clusterings by adding new clusterings to the set based on maximizing the minimum pairwise distance to every other clustering currently in the set. This process is seeded with the informationally maximal pair within the set. This does not avoid triple-wise information overlap but works well in practice.

5 Physics Articles from arXiv.org

ArXiv.org releases convenient metadata (title, authors, etc...) for all articles in their database. Additionally, a special set of 30 000 high energy physics articles are released with abstracts and citation networks. We apply our process to this network of papers with edge types *Titles, Authors Abstracts and Citations*.



(a)

(b)

Fig. 5. (a) The pairwise distances between the sampled clusterings form a graph. Note the dark blocks along the diagonal. These are indicative of tightly knit clusters. (b) A dendrogram of this graph. We use the ordering of the vertices picked out by the dendrogram to optimally highlight the blocks in the left image.

Articles are connected by *title* or *abstract* based on the cosine similarity of the text (using the bag of words model[1]). Two articles are connected by *author* by the number of authors that the two articles have in common. Two articles are connected by *citation* if either article cites the other (undirected). We inspect this system with the following process discussed in greater detail above.

These graphs are normalized by the L_2 norm and then the space of composite edge types is sampled uniformly. That is $\omega_j = \sum_{i=1}^{4} \alpha_i w_i$, where $\alpha_i \in (-1, 1)$, $w_i \in \{$*titles, abstract, authors, citation*$\}$. The resulting graphs are then clustered using Clauset et al's FastModularity[2] algorithm. The resulting clusterings are compared in a graph which is then clustered to produce clusters of clusterings. The clusters of clusterings are averaged [14] and we inspect the resultant representative-clusterings.

The similarity matrix of the graph of clusterings is shown in Fig. 5(a). The presence of blocks on the diagonal imply clusters of clusterings. From this process we obtain representative-clusterings. The various partitionings of the original set of papers vary considerably (large VI distance) yet exhibit high modularity scores implying a variety of high-quality clusterings within the dataset.

Table 1. Commonly appearing words (stemmed) in two distinct representative-clusterings. Clusters within each clustering correspond to well known subfields in High-Energy Physics (subfield 1,2,3,4,5 will replace these with actual names in a bit). This data however does not show a strong distinction between the clusterings. Furher investigation is warranted.

Cluster	Statistically Significant Words in Clustering 1
1	quantum, algebra, integr, equat, model, chern-simon, lattic, particl, affin
2	potenti, casimir, self-dual, dilaton, induc, cosmolog, brane, anomali, scalar
3	black, hole, brane, supergrav, cosmolog, ads/cft, sitter, world, entropi
4	cosmolog, black, hole, dilaton, graviti, entropi, dirac, 2d, univers
5	d-brane, tachyon, string, matrix, theori, noncommut, dualiti, supersymmetr, n=2

Cluster	Statistically Significant Words in Clustering 2
1	potenti, casimir, self-dual, dilaton, induc, energi, scalar, cosmolog, gravit
2	integr, model, toda, equat, function, fermion, casimir, affin, dirac
3	tachyon, d-brane, string, orbifold, n=2, n=1, dualiti, type, supersymmetr
4	black, hole, noncommut, supergrav, brane, sitter, entropi, cosmolog, graviti

Analysis of this dataset is challenging and still in progress. We can look at articles in a clustering and inspect attributes like the country (by submitting e-mail's country code), or words which occur more often than statistically expected given the corpus. Most clusterings found show a separation into various topics identifyable by domain experts (example in Table 1) however a distinction between clusterings has not yet been found. While the VI distance between metaclusterings presented in Fig. 5(a) is large it has so far proven difficult to identify the qualitative distinction for the quantitative difference. More in depth inspection by a domain expert may be necessary.

6 Conclusion and Future Work

We investigated clustering in the context of network data with multiple relationships between nodes. We found that a rich clustering structure can exist with clusters of clusterings. In an example we found that by reducing this clustering structure we uncovered latent classes which explained the underlying graph very compactly. We presented a simple method that works well on simple cases.

In the future it will be interesting to apply these methods to more challenging problems and see which aspects become interesting. There is much room for growth in this topic. Ongoing work includes more intelligent sampling (intentionally finding distinct clusterings), effects of adding non-linear combinations of edge-types, and searching the space for clusterings with desired attributes.

References

1. David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(4-5):993–1022, May 2003.
2. Aaron Clauset, M E J Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 70(6 Pt 2):066111, December 2004.
3. Inderjit S Dhillon, Yuqiang Guan, and Brian Kulis. Weighted graph cuts without eigenvectors a multilevel approach. *IEEE transactions on pattern analysis and machine intelligence*, 29(11):1944–57, November 2007.
4. Daniel M. Dunlavy, Tamara G. Kolda, and W. Philip Kegelmeyer. Multilinear Algebra For Analyzing Data With Multiple Linkages, 2006.
5. Stephen E. Fienberg, Michael M. Meyer, and Stanley S. Wasserman. Statistical Analysis of Multiple Sociometric Relations. *Journal of the American Statistical Association*, 80(389):51 – 67, 1985.
6. Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: A comparative analysis. *Physical Review E*, 80(5), November 2009.
7. T. Lange, V. Roth, M. L. Braun, and J. M. Buhmann. Stability-based validation of clustering solutions, neural computation. *Neural Computation*, 16:1299–1323, 2004.
8. X. Luo. On coreference resolution performance metrics. In *Proc. Human Language Technology Conf. and Conf. Empirical Methods in Natural Language Processing*, pages 25–32, Vancouver, British Columbia, Canada, 2005. Association for Computational Linguistics.
9. Marina Meila. Comparing Clusterings by the Variation of Information. *Technical Report*, pages 173–187, 2003.
10. B. Mirkin. *Mathematical Classification and Clustering*. Kluwer Academic Press, 1996.
11. P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J. P. Onnela. Community Structure in Time-Dependent, Multiscale, and Multiplex Networks. *Science*, 328(5980):876–878, May 2010.
12. Matthew Rocklin and Ali Pinar. Computing an aggregate edge-weight function for clustering graphs with multiple edge types. In *Proc. 7th Workshop on Algorithms and Models for the Web Graph (WAW10)*, 2010.
13. C. Stichting, M. Centrum, and S. V. Dongen. Performance criteria for graph clustering and markov cluster experiments. Technical Report INS-R0012, Centre for Mathematics and Computer Science, 2000.
14. Alexander Strehl and Joydeep Ghosh. Cluster EnsemblesA Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*, 3(3):583–617, March 2003.